

Celem ćwiczenia jest zapoznanie się z dziedziczeniem i zastosowaniem polimorfizmu. Do zajęć wymagane jest posiadanie kodu z ćwiczenia 3. Na zajęciach należy wykonać następujące zadania:

1. Stworzyć interfejs klasy IFile (tylko plik IFile.h). Zawrzeć w nim metody z klas CsvFile i BinaryFile: read(vector<Point>&), read(Point, int), write(vector<Point>&). Metody w klasie IFile muszą być czysto wirtualne.
2. Przenieść prywatne pola z klas CsvFile i BinaryFile do klasy IFile.
3. Zmienić klasy CsvFile i BinaryFile, tak aby dziedziczyły po klasie IFile
4. Stworzyć klasę FileFactory, która będzie zawierać tylko jedną metodę statyczną OpenFile, która przyjmuje argumenty:
  - path – łańcuch znaków określający ścieżkę do pliku,
  - openMode – tryb otwarcia pliku,

Metoda OpenFile będzie zwracać wskaźnik na IFile. Metoda powinna rozpoznać czy otwieramy plik CSV czy plik binarny i w zależności od tego utworzyć dynamicznie nowy obiekt odpowiedniej klasy.

5. Przetestować działanie metody CreateFile otwierając pliki o różnych formatach.

#### Wskazówki:

1. Zapoznać się z mechanizmem dziedziczenia w języku c++.
2. Zapoznać się z mechanizmem metod czysto wirtualnych:  
[https://en.cppreference.com/w/cpp/language/abstract\\_class](https://en.cppreference.com/w/cpp/language/abstract_class)
3. Należy zapoznać się z przykładowym programem wykorzystującym dziedziczenie (Kod 4.1)

#### Uwagi:

- W metodach klas nie wolno używać funkcji typu : printf, scanf, std::cout, std::cin. Warstwa prezentacji musi zostać zaimplementowana w pętli głównej (poza klasą),
- W plikach nagłówkowych (\*.h) nie wolno zawierać źródeł/rozwinień funkcji i metod klasowych (dotyczy to także konstruktorów i destruktorów),
- Powyższe uwagi obowiązują na wszystkich laboratoriach.